

**Amendments to the Claims:**

The listing of claims will replace all prior versions and listings of claims in the application:

**Listing of Claims:**

- 5 Claim 1 (previously presented): A method for program debugging, the method comprising:
- setting a plurality of breakpoints corresponding to a plurality of events in a Basic Input/Output System (BIOS) program code, each event being a test executed by the BIOS program code to a peripheral device and taking a general
- 10 processing path when the peripheral device is working well or an error processing path when the peripheral device is in an error state;
- executing the BIOS program code for outputting a diagnosis code of a breakpoint;
- setting a parameter to simulate the peripheral device is working well throughout
- 15 execution of the event corresponding to the diagnosis code;
- executing the event corresponding to the diagnosis code according to the parameter for making the event undergo the general processing path;
- resetting the parameter to simulate the peripheral device being in the error state throughout execution of the event corresponding to the diagnosis code; and
- 20 executing the event corresponding to the diagnosis code according to the reset parameter for making the event undergo the error processing path.

Claim 2 (canceled)

- 25 Claim 3 (original): The method of claim 1 wherein the breakpoints are set ahead of program codes of the corresponding events.

Claim 4 (original): The method of claim 1 wherein the breakpoints are set after program codes of the corresponding events.

30

Claim 5 (canceled)

Claim 6 (previously presented): The method of claim 1 wherein the error processing path produces an audible tone.

5

Claim 7 (previously presented): The method of claim 1 wherein the error processing path causes a system reset.

10 Claim 8 (previously presented): The method of claim 1 wherein the error processing path causes a system execution interrupt.

Claims 9-16 (canceled)

15 Claim 17 (previously presented): The method of claim 1 further comprising:  
executing the BIOS program code until the diagnosis code of the breakpoint matches a predetermined diagnosis code before resetting the parameter of the event corresponding to the diagnosis code, and executing the event corresponding to the diagnosis code according to the reset parameter for making the event undergo the error processing path.

20 Claim 18 (previously presented): A method for program debugging, the method comprising:  
setting a plurality of breakpoints corresponding to a plurality of events in a driver program code, each event being a test executed by the driver program code to a peripheral device and taking a general processing path when the peripheral device is working well or an error processing path when the  
25 peripheral device is in an error state;  
setting a parameter to simulate that the peripheral device is working well throughout execution of the driver program code;  
executing the driver program code according to the parameter for outputting a diagnosis code corresponding to each breakpoint;  
30 for each breakpoint, determining whether the diagnosis code matches a user

defined diagnosis code; and  
resetting the parameter to simulate that the peripheral device is in the error state  
and executing the event corresponding to the diagnosis code according to  
the reset parameter for making the event undergo the error processing path  
5 when it is determined that the diagnosis code matches the user defined  
diagnosis code.

Claim 19 (previously presented): The method of claim 18 further comprising  
continuing execution of the driver program code to a next breakpoint without  
10 resetting the parameter when it is determined that the diagnosis code does not  
match the user defined diagnosis code.

Claim 20 (canceled)

15 Claim 21 (previously presented): The method of claim 1 further comprising executing  
both the general processing path and the error processing path of all events of the  
plurality of events.

Claim 22 (previously presented): A method for program debugging, the method  
20 comprising:  
setting a plurality of breakpoints corresponding to a plurality of events in a Basic  
Input/Output System (BIOS) program code, each event being a test executed  
by the BIOS program code to a peripheral device and taking a general  
processing path when the peripheral device is working well and taking a  
25 generic event error handling path or a critical error path when the peripheral  
device is in an error state according to the error state, the path taken  
determined by a parameter;  
setting the parameter to determine the general processing path;  
executing the BIOS program code according to the parameter for outputting a  
30 diagnosis code at each of the breakpoints, each diagnosis code uniquely

- indicating the event corresponding to the breakpoint;  
when the outputted diagnostic code matches a predetermined diagnostic code,  
resetting the parameter to determine which of the generic event error  
handling path or the critical error path is to be taken;
- 5      when the reset parameter determines the generic event error handling path is to  
be taken, executing the BIOS program code according to the reset parameter;  
and  
when the reset parameter determines the critical event error handling path is to be  
taken, executing the BIOS program code according to the reset parameter.
- 10      Claim 23 (previously presented): The method of claim 22 further comprising when  
executing the BIOS program code according to the reset parameter and the reset  
parameter determines the generic event error handling path is to be taken, writing  
error messages to a file.
- 15      Claim 24 (previously presented): The method of claim 22 further comprising when  
executing the BIOS program code according to the reset parameter and the reset  
parameter determines the critical event error handling path is to be taken, the  
critical error handling path generates an audible tone, a system reset, or a stop  
20      execution command.